

УДК 004.94:004.021

М.И. Кочергин

Компенсация амплитудно-временной погрешности в дискретно-непрерывных компонентных моделях на основе задачи обратной интерполяции

Рассматриваются подходы к решению задачи корректного обнаружения событий в гибридных (дискретно-непрерывных системах) и снижения возникающей накапливающейся погрешности по величине целевой (отслеживаемой) переменной и по времени. Предлагается численный алгоритм компенсации такой амплитудно-временной погрешности, основанный на решении обратной задачи интерполяции. Сравнивается результативность предлагаемого метода и метода, заложенного в гибридные автоматы. Оцениваются возможности предлагаемого метода для согласования вектора состояния компонентных моделей подсистем моделируемого объекта, которые представлены на различных слоях объектного уровня многоуровневой компьютерной модели, при проведении вычислительного эксперимента на модели.

Ключевые слова: моделирование, амплитудно-временная погрешность, метод компонентных цепей, гибридные системы, конечный автомат, диаграмма состояний.

DOI: 10.21293/1818-0442-2024-27-3-79-84

При исследовании сложных динамических систем исследователи [1, 2] выделяют класс гибридных (дискретно-непрерывных) систем – абстракций для описания реальных систем в виде моделей, характеризующихся сменой своего непрерывного поведения при наступлении определенных событий. Такие модели имеют дискретную составляющую, выраженную во множестве допустимых состояний и условиях перехода, которую удобно моделировать конечными автоматами, гибридными автоматами [3] или диаграммами состояний [4], и непрерывную, моделируемую, как правило, системами дифференциальных уравнений. Такие системы также могут называться дискретно-событийными.

При моделировании таких систем возникает задача корректного обнаружения событий гибридных систем, решаемая как зарубежными [5–8], так и отечественными [9, 10] исследователями. Предлагаемые исследователями решения могут быть сгруппированы в 3 категории:

1) варьирующие (замедляющие) шаг моделирования (интегрирования) при приближении к границе смены режима гибридной системы;

2) решающие итерационными методами нелинейные уравнения вида $f(t) = 0$, описывающего кривую поведения в пространстве состояния, для определения момента времени t пересечения границы смены режима гибридной системы. Такое решение делится на 2 фазы [11, 12]: обнаружение перехода с пробным расчётом новой итерации и локализация точки смены режима с применением численного метода;

3) методы, основанные на аппроксимации функции поведения системы и её исследовании, в том числе с применением прогнозных методов.

Недостатком методов первой группы является негативное влияние на быстродействие анализа (расчёта) компьютерной модели, а большинства методов второй группы – риски пропуска точек пере-

ключения на немонотонных функциях поведения (хотя существуют алгоритмы [2], позволяющие обнаруживать многократные переходы ценой больших вычислительных затрат). Численные характеристики сравнения различных вычислительных методов приведены в [13].

В данной работе предлагается алгоритм обнаружения точки перехода и компенсации возникающей амплитудно-временной погрешности на основе решения обратной задачи интерполяции.

Природа амплитудно-временной погрешности

Применение диаграмм состояний или конечных автоматов для описания поведения систем сопряжено с выполнением на каждой итерации проверки значения целевой переменной, выход которой за пределы диапазона, заданного в сторожевом условии перехода, инициирует смену режима. Допустим, что система должна перейти в новое состояние при выходе системы $y = 0$ (в виду дискретизации модели условие корректно формулировать как $y \leq 0$). Тогда срабатывание условия перехода произойдёт при $y = -\Delta y$ (в случае изначального уменьшения значения y), где Δy – некоторое малое число (погрешность) – амплитудная составляющая погрешности. При этом (рис. 1) из-за смещения y ниже оси ОХ переход в новое состояние произойдёт позже на Δt – временная составляющая погрешности.

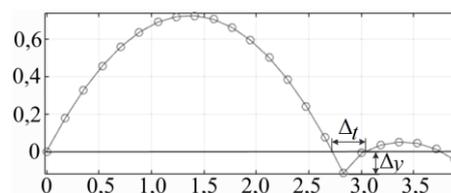


Рис. 1. Иллюстрация амплитудной и временной составляющей погрешности при моделировании отскока тела от поверхности

Источник погрешности – особенности машинных вычислений, осуществляемых с дискретным

шагом. Несмотря на то, что величина погрешности является незначительной, она носит аккумулятивный характер с течением времени при достаточно большом количестве переходов из одного состояния в другое. При построении модели штангового глубинного насоса [14] допустимо применение диаграмм состояний [4] для задания смены направления движения штока при возвратно-поступательном

движении. При этом можно наблюдать эффект смещения зависимости характеристик с течением времени. На рис. 2 приведено смещение зависимости скорости штока штангового глубинного насоса от времени с течением времени. Штрихпунктирной линией показана зависимость, полученная при анализе модели с шагом интегрирования 0,001 с, сплошной – 0,01 с.

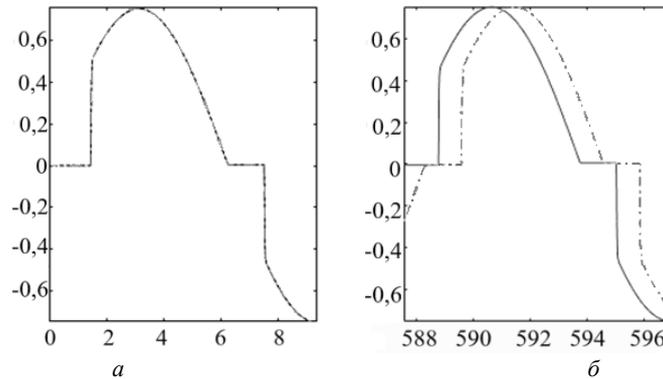


Рис. 2. Смещение графика зависимости скорости штока штангового глубинного насоса от времени: a – на интервале $t = [0, 10]$ с; b – на интервале $t = [588, 598]$ с

Уменьшить рассматриваемую погрешность можно за счёт уменьшения шага моделирования (интегрирования). Но полностью избавиться от такой погрешности при стандартном подходе не получится – со временем она продолжит накапливаться в модели, что может сильно повлиять на корректность результатов моделирования, особенно моделей, предназначенных для длительного расчёта, например моделей цифровых двойников [15].

Постановка задачи

Сформулируем постановку задачи для решения предлагаемый алгоритмом. Имеется динамическая система с гибридным поведением, которая в один момент времени находится в одном из дискретных состояний S_i , $S = \{S_0, S_1, \dots, S_n\}$, где каждому S_i установлено соответствие $R = S \rightarrow C, R \subseteq S \times C$ компонентной схеме (цепи) C_j из общего множества компонентных схем модели $C = \{C_0, C_1, \dots, C_m\}$, описывающей в данном дискретном состоянии непрерывное поведение системы (1), где $f_i(y, t)$ – вектор-функция правых частей дифференциальных уравнений (каждая компонентная цепь является эквивалентом системы алгебро-дифференциальных уравнений первого порядка вида (1), записанных относительно переменных состояния y), t_0 – время начала очередного дискретного состояния (и завершения предыдущего), t_i – время завершения очередного дискретного состояния (и начала следующего), y_0 – вектор начальных условий для дифференциальных переменных (конечное значение y предыдущего состояния является начальным условием следующего).

$$\frac{dy}{dt} = f_i(y, t), \quad y(t_0) = y_0, \quad t \in [t_0, t_i], \quad i = 0, \dots, n. \quad (1)$$

Порядок переходов (переход считается мгновенным) между дискретными состояниями S определяется множеством дискретных переходов

$T = \{(S_0, S_1), \dots, (S_{n-1}, S_n)\}$, $T \subseteq S \times S$. Переход из S_i в S_j осуществляется по сторожевому условию, задаваемому соответствующим предикатом $G_{ij}(y_{ц})$ вида $y_{ц} \geq a$ или $y_{ц} \leq a$, отслеживающим изменение целевой переменной $y_{ц}$, где $y_{ц} \in y$, a – некоторая константа. В каждом дискретном состоянии S может быть своя целевая переменная $y_{ц}$, а вместо константы a может задаваться аналитическое выражение, подлежащее расчёту.

При интегрировании дифференциальных уравнений вида (1) ввиду дискретности шага интегрирования срабатывание предиката $G_{ij}(y_{ц})$ (а вследствие этого и переход в новое дискретное состояние) произойдет в момент времени $t = t^* + \Delta t$ при $y_{ц} = a \pm \Delta a$ (знак зависит от роста или убывания $y_{ц}$), где Δa – амплитудная составляющая погрешности, Δt – временная, t^* – реальное (искомое) время перехода.

Наличие такой погрешности в модели системы является нежелательным (а при некоторых условиях неприемлемым) по причине её аддитивности (накопления с каждой сменой дискретного состояния) и неправильного обнаружения точки переключения одной непрерывной модели поведения системы на другую. Цель работы алгоритма – определение момента времени t^* , при котором выполнялось условие $y_{ц} = a$, и пересчёт вектора решения (включающего переменные x, y исследуемой системы и переменные z внешней по отношению к исследуемой системе среды) компонентной схемы с переносом непрерывного состояния системы в пространстве состояний из (y, t) в (y^*, t^*) и внешней среды из (z, t) в (z^*, t^*) , где $z \in Z$, Z – вектор переменных состояния внешней среды. К внешней среде исследуемой системы (подсистемы) следует также относить другие подсистемы системы верхнего уровня, функционирующие совместно с исследуемой.

Алгоритм компенсации амплитудно-временной погрешности

Рассмотрим работу алгоритма компенсации амплитудно-временной погрешности на примере. Переход системы из состояния 1 в состояние 2 осуществляется при условии $f(t) = 0$. Перед моментом ожидаемого переключения имеем $f(t_i) = +\Delta y$, а после расчёта ещё одной итерации: $f(t_{i+1}) = -\Delta y$. Предлагаемый в работе алгоритм, получив сигнал об инициализации перехода в новое состояние, осуществляет перерасчёт всех переменных вектора решений модели по интерполяционной формуле Лагранжа (2) [16], тем самым «откатывая» состояние (вектор решения) системы в сторону предыдущей итерации до точки $f(t_{i+1}) = 0$, после чего расчёт модели продолжается с рассчитанных значений вектора решений.

$$x = \sum_{k=0}^n \frac{\prod_{i \neq k} (y - y_i)}{\prod_{i \neq k} (y_k - y_i)} \cdot x_k, \quad (2)$$

где x – узел интерполяции, y – значение функции в соответствующем узле.

Алгоритм предусматривает следующие варианты дальнейших расчётов и изменения расчётной сетки [17] (рис. 3):

а) однократное изменение шага и продолжение расчёта по смещённой сетке с прежним шагом dt (рис. 4: не выполняются условия «Смещение текущей точки назад» и «Возврат к старой сетке»);

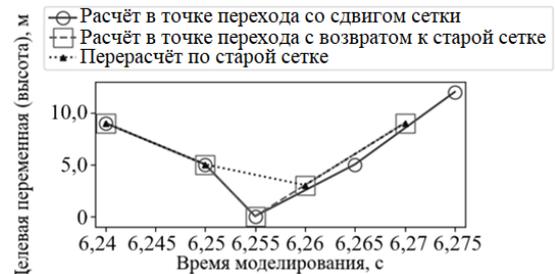


Рис. 3. Иллюстрация вариантов изменения расчётной сетки в результате работы алгоритма

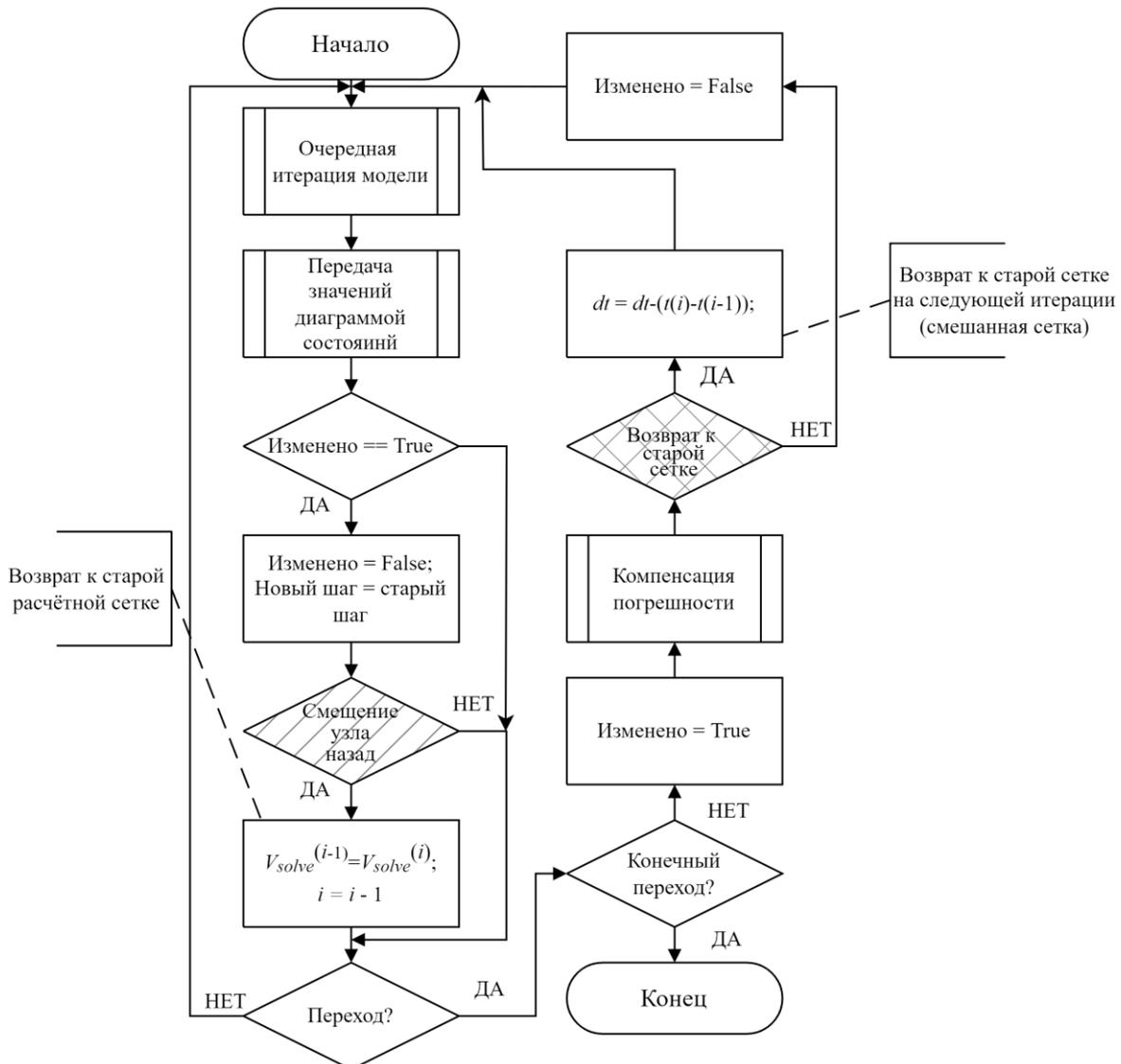


Рис. 4. Блок-схема алгоритма компенсации амплитудно-временной погрешности

б) добавление дополнительного узла сетки в окрестности точки перехода системы в новое состояние и продолжение последующих расчётов в узлах старой сетки (рис. 4: выполняется только условие «Возврат к старой сетке»);

в) нахождение точки перехода системы в новое состояние и перерасчёт вектора решения в следующем узле старой сетки без изменений её узлов и добавления новых (выполняются оба условия).

Наличие трёх вариантов изменения сетки позволяет организовать согласование вектора решения компонентных моделей подсистем, представленных на разных уровнях объектного уровня (т.е. представленными изолированными компонентными моделями) в многоуровневой компьютерной модели [18], в соответствии с заданными целями. Так, смещение расчётной сетки (вариант *a*) применимо в моделях с независимыми объектными слоями, рассчитываемыми разными вычислительными ядрами, при необходимости исследования экстремальных значений целевой переменной в рамках вычислительного эксперимента и при наличии требований к равной длине массива траектории поведения подсистем (т.е., когда необходимо, чтобы в результатах моделирования каждой подсистемы было одинаковое количество записей (точек)).

Добавление дополнительного узла (вариант *b*) применимо в тех же условиях, но при отсутствии требования к длине массивов результатов моделирования. Перерасчёт по старой сетке (вариант *в*) применим в случаях, когда результаты расчёта компонентной цепи на одном слое используются в параллельном расчёте другой компонентной цепи на другом слое – т.е. тогда, когда смещение расчётной сетки для одной подсистемы может привести к погрешностям в расчёте другой, а одновременный пересчёт сетки во всех слоях модели может привести к циклическому срабатыванию условий перехода в новое состояние каких-либо подсистем и, как следствие, к повышению вычислительных затрат.

Вызываемая на рис. 4 процедура «Компенсация погрешности» рассчитывается по формуле (2) для всего вектора решений без смены счётчика итераций, процедура «Очередная итерация модели» выполняет расчёт новой итерации вектора решений компонентной модели, процедура «Передача значений диаграммой состояний» инициализирует работу

текущего компонента «Событие» или «Состояние», которая заключается в передаче определённого на схеме модели значения параметра на выход компонента с целью дальнейшей передачи сообщения для параметризации компонента объектного или логического слоя. Более подробно механизм работы диаграмм состояний метода компонентных цепей описан в работе [4].

Оценка результатов и обсуждение

В работе [19] при использовании гибридных автоматов для моделирования гибридной механической системы предлагается присвоить целевой переменной $f(t)$ в момент инициализации перехода системы в новое состояние значение 0 с целью устранения погрешности рассматриваемого типа. Однако, по нашему мнению, данная процедура не только не снизит погрешность, но, наоборот, приведёт к рассогласованию непрерывного состояния системы в целом и её подсистем и росту погрешности, так как компенсация погрешности при таком подходе осуществляется только для одной переменной, а не всего вектора решения.

Продemonстрируем это на примере сравнения с эталонной моделью моделей, построенных с шагом моделирования $dt = 10^{-3}$ с. В качестве эталона возьмём модель без компенсации погрешностей большей точности – с шагом моделирования $dt = 10^{-6}$ с.

В таблице представлено сравнение трёх моделей:

- 1) использующей предложенный в данной работе алгоритм компенсации погрешности,
- 2) использующей алгоритм гибридного автомата [19],
- 3) не использующей алгоритмы компенсации погрешности.

Сравнение будет производиться по целевой переменной $f(t)$ и её производной $df(t)/dt$, чтобы проиллюстрировать эффект взаимного усиления погрешностей – ошибка в расчёте $f(t)$ влечёт за собой возникновение ошибки и в $df(t)/dt$ при решении системы дифференциальных уравнений, что, в свою очередь, влияет и на расчёт $f(t)$ на новой итерации. Для выполнения сравнения результатов моделирования с разным шагом интегрирования из табличных зависимостей, построенных с шагом $dt = 10^{-6}$ с, были удалены промежуточные узлы, в которых отсутствовали значения переменных в отклике какой-либо из моделей (таблица).

Сравнение погрешности моделей при применении предлагаемого алгоритма, алгоритма-аналога и без применения алгоритмов компенсации погрешности

| Время t оценки погрешности | Модель при $dt = 10^{-3}$ с | Среднее абсолютное отклонение от результатов моделирования при $dt = 10^{-6}$ с | | Среднее относительное отклонение от результатов моделирования при $dt = 10^{-6}$ с | |
|---|-----------------------------|---|-------------------|--|-------------------|
| | | $\Delta f(t)$ | $\Delta df(t)/dt$ | $\delta f(t)$ | $\delta df(t)/dt$ |
| В момент первого перехода в новое состояние | Алгоритм компенсации | 0,002938 | 0,02443 | 0,693% | 1,384% |
| | Гибридный автомат | 0,002940 | 0,02444 | 0,693% | 1,384% |
| | Без компенсации | 0,002942 | 0,02444 | 0,694% | 1,384% |
| В момент второго перехода в новое состояние | Алгоритм компенсации | 0,002922 | 0,04957 | 2,760% | 2,8% |
| | Гибридный автомат | 0,003634 | 0,06119 | 3,432% | 3,5% |
| | Без компенсации | 0,003282 | 0,05654 | 3,100% | 3,2% |

В таблице выделены полужирным наименьшие значения погрешностей и подчеркнуты первые отличающиеся в них значащие цифры.

Как видно из сопоставления таблиц, после третьего дискретного состояния общая погрешность решения без использования алгоритма компенсации выросла с $\delta f(t) = 0,69\%$ до $\delta f(t) = 3,10\%$, а с использованием – всего до $\delta f(t) = 2,76\%$. Эта разница (0,34%) и представляет собой амплитудно-временную погрешность, которую компенсирует предложенный алгоритм (после первого перехода в новое состояние разница составляла 0,001%, что свидетельствует о её росте с течением времени). Общий рост погрешности (с 0,69 до 2,76%) связан с разницей шага моделирования (интегрирования) рассматриваемой модели и эталонной.

Заключение

Предложенный в данной работе алгоритм компенсации амплитудно-временной погрешности основан на решении обратной задачи интерполяции, что обеспечивает снижение накапливаемой в конечных автоматах и диаграммах состояний погрешности без привлечения итерационных процедур, характерных для численных методов решения уравнений или методов адаптации шага моделирования. Требуемая гладкость функции состояния системы может быть обеспечена повышением степени полинома для кусочной интерполяции, что влечёт за собой снижение быстродействия алгоритма.

Параметрируемость алгоритма обеспечивает его применимость как в задачах проведения вычислительного эксперимента, требующего точных оценок экстремальных значений переменных в модели, так и в моделях сложных систем, представленных распределёнными компонентными цепями на разных слоях объектного уровня и требующих согласованности узлов расчёта вектора решения во временной области.

Литература

1. Колесов Ю.Б. Развитие метода объектно-ориентированного анализа для задач проектирования гибридных систем управления: дис. ... д-ра техн. наук. – СПб., 2003. – 252 с.
2. Сениченков Ю.Б. Основы теории и средства моделирования гибридных систем: дис. ... д-ра техн. наук. – СПб., 2005. – 312 с.
3. Колесов Ю.Б. Моделирование систем. Динамические и гибридные системы / Ю.Б. Колесов, Ю.Б. Сениченков. – СПб.: БХВ-Петербург, 2012. – 224 с.
4. Kochergin M.I. Interpretation of the statechart diagram into a multilevel simulation language // Proceedings of the TUSUR University. – 2017. – Vol. 20, No. 4. – P. 122–125. DOI: 10.21293/1818-0442-2017-20-4-122-125.
5. Mosterman P. An overview of hybrid simulation phenomena and their support by simulation packages // Hybrid Systems: Computation and Control: vol. 1569 of Lecture Notes in Computer Science. – Springer Verlag, 1999. – P. 165–177.
6. Park T. Stale event location in differential-algebraic models. ACM Transactions on Modeling and Computer Simulation / T. Park, P.I. Barton // TOMACS. – 1996. – № 6 (2). – P. 137–165.

7. Cellier F. Combined discrete continuous system simulation by use of digital computers: techniques and tools // PhD thesis, ETH Zurich. – Switzerland, 1979. – P. 144–156.

8. Gear C.W. Solving ordinary differential equations with discontinuities / C.W. Gear, O. Osterby // Technical report. – Dept. of Comput. Sci., University of Illinois, 1981. – P. 27–31.

9. Сениченков Ю.Б. Численное моделирование гибридных систем. – СПб.: Изд-во Политехн. ун-та, 2004. – 206 с.

10. Шорников Ю.В. Прикладное математическое, алгоритмическое и программное обеспечение компьютерного анализа гибридных систем: дис. ... д-ра техн. наук: 05.13.11. – Новосибирск, 2009. – 313 с.

11. Bruck D. Dyraola user's manual / D. Bruck, H. Elmquist, M. Otter. – Lund Switzerland, 1996. – 340 p.

12. A hierarchical hybrid system model and its simulation / J. Lui, X. Lui, T.J. Koo, B. Sinopoli, S. Sastry, E.A. Lee // Proceedings of the 38th Conference on Decision and Control. – 1999. – P. 2407–2411.

13. Попов Е.А. Детекция событий разного типа в гибридных динамических системах / Е.А. Попов, Ю.В. Шорников // Научный вестник Новосиб. гос. техн. ун-та. – 2020. – № 4 (80). – С. 159–176. DOI: 10.17212/1814-1196-2020-4-159-176.

14. Многоуровневое моделирование физико-технических задач на примере глубинного насоса / М.И. Кочергин, Д.С. Торгаева, М.П. Сухоруков, В.М. Дмитриев, Ю.А. Шурыгин // Доклады ТУСУР. – 2019. – Т. 22, № 3. – С. 79–86. DOI: 10.21293/1818-0442-2019-22-3-79-86.

15. Дмитриев В.М. Реализация цифрового двойника в среде моделирования MAPC / В.М. Дмитриев, Л.А. Гембух, А.Е. Сахабудинов // Вестник Южно-Уральского гос. ун-та. – Сер.: Компьютерные технологии, управление, радиоэлектроника. – 2023. – Т. 23, № 4. – С. 26–36. DOI: 10.14529/ctcr230403.

16. Numerical Recipes in C. The Art of Scientific Computing / W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery. – Cambridge: Cambridge University Press, 1997. – 1018 p.

17. Thompson J.F. Numerical Grid Generation, Foundations and Applications / J.F. Thompson, Z.A. Warsi, C.V. Martin. – Amsterdam: North-Holland, 1985. – 331 p.

18. Дмитриев В.М. Матрично-топологический анализ компонентных цепей / В.М. Дмитриев, Т.В. Ганджа, М.И. Кочергин // Вестник Том. гос. ун-та. Управление, вычислительная техника и информатика. – 2023. – № 62. – С. 25–35.

19. Колесов Ю.Б. Объектно-ориентированное моделирование в среде Rand Model Designer 7: учеб.-практ. пособие / Ю.Б. Колесов, Ю.Б. Сениченков. – М.: Проспект, 2016. – 256 с.

Кочергин Максим Игоревич

Канд. техн. наук, доцент каф. компьютерных систем в управлении и проектировании (КСУП)
Томского государственного университета
систем управления и радиоэлектроники (ТУСУР)
Ленина пр-т, 40, г. Томск, Россия, 634050
ORCID: 0000-0001-7404-8718
Тел.: 8 (382-2) 70-15-15, внутр.: 20-64
Эл. почта: maksim.i.kochergin@tusur.ru

Kochergin M.I.

Compensation of amplitude-time error in discrete-continuous component models based on the inverse interpolation problem

The paper discusses approaches to solve the problem of correct event detection in hybrid (discrete-continuous systems) and reducing the resulting cumulative error in the value of the target (tracked) variable and in time. A numerical algorithm for compensating for such an amplitude-time error is proposed, based on solving the inverse interpolation problem. The efficiency of the proposed method and the method embedded in hybrid automata is compared. The capabilities of the proposed method to match the state vector of component models of the subsystems of the simulated object, presented on different layers of the object level of a multi-level computer model, are assessed when conducting a computational experiment on the model.

Keywords: modeling, amplitude-time error, component circuit method, hybrid systems, finite state machine, state diagram.

DOI: 10.21293/1818-0442-2024-27-3-79-84

References

1. Kolesov Yu.B. *Razvitie metoda ob'ektno-orientirovannogo analiza dlya zadach proektirovaniya gibridnyh sistem upravleniya* [Development of the method of object-oriented analysis for the problems of designing hybrid control systems. Dissertation for Doctor of Science in Engineering title]. St. Petersburg, 2003, 252 p. (in Russ.).
2. Senichenkov Yu.B. *Osnovy teorii i sredstva modelirovaniya gibridnyh sistem* [Fundamentals of the theory and means of modeling hybrid systems. Dissertation for Doctor of Science in Engineering title]. St. Petersburg, 2005, 312 p. (in Russ.).
3. Kolesov Yu.B., Senichenkov Yu.B. *Modelirovanie sistem. Dinamicheskie i gibridnye sistemy* [Modeling of systems. Dynamic and hybrid systems]. St. Petersburg: BHV-Petersburg, 2012, 224 p. (in Russ.).
4. Kochergin M.I. Interpretation of the statechart diagram into a multilevel simulation language, *Proceedings of TUSUR University*, 2017, vol. 20, no. 4, pp. 122–125. DOI: 10.21293/1818-0442-2017-20-4-122-125.
5. Mosterman P. An overview of hybrid simulation phenomena and their support by simulation packages, *Hybrid Systems: Computation and Control*, vol. 1569 of Lecture Notes in Computer Science, Springer Verlag, 1999, pp. 165–177.
6. Park T., Barton P.I. Stale event location in differential-algebraic models. *ACM Transactions on Modeling and Computer Simulation, TOMACS*, 1996, no. 6 (2), pp. 137–165.
7. Cellier F. *Combined discrete continuous system simulation by use of digital computers: techniques and tools*. PhD thesis, ETH Zurich, Switzerland, 1979, pp. 144–156.
8. Gear C.W., Osterby O. Solving ordinary differential equations with discontinuities, *Technical report*, Dept. of Comput. Sci., University of Illinois, 1981, pp. 27–31.
9. Senichenkov Yu.B. *CHislennoe modelirovanie gibridnyh sistem* [Numerical modeling of hybrid systems]. St. Pe-

tersburg: Publishing house of the Polytechnic University, 2004, 206 p. (in Russ.).

10. Shornikov Yu.V. *Prikladnoe matematicheskoe, algoritmicheskoe i programnoe obespechenie kompyuternogo analiza gibridnyh sistem* [Applied mathematical, algorithmic and software support for computer analysis of hybrid systems. Dissertation for Doctor of Science in Engineering title]. Novosibirsk, 2009. 313 p. (in Russ.).

11. Bruck D., Elmqvist H., Otter M. *Dyraola user's manual*. Switzerland, Lund, 1996, 340 p.

12. Lui J., Lui X., Koo T.J., Sinopoli B., Sastry S., Lee E.A. A hierarchical hybrid system model and its simulation, *Proceedings of the 38th Conference on Decision and Control*, 1999, pp. 2407–2411.

13. Popov E.A., Shornikov Yu.V. Detection of different type events in hybrid dynamical systems, *Science Bulletin of the Novosibirsk State Technical University*, 2020, no. 4 (80), pp. 159–176. DOI: 10.17212/1814-1196-2020-4-159-176 (in Russ.).

14. Kochergin M.I., Torgaeva D.S., Sukhorukov M.P., Dmitriev V.M., Shyrigin Yu.A. Multilevel modeling of physical and technical problems using the example of a deep-well pump, *Proceedings of TUSUR University*, 2019, vol. 22, no. 3, pp. 79–86. DOI 10.21293/1818-0442-2019-22-3-79-86 (in Russ.).

15. Dmitriev V.M., Gembukh L.A., Sakhabutdinov A.E. Implementation of a Digital Twin in the MARS Modeling Environment, *Bulletin of the South Ural State University. Series: Computer Technologies, Control, Radio Electronics*, 2023, vol. 23, no. 4. pp. 26–36. DOI 10.14529/ctcr230403 (in Russ.).

16. Press W.H., Teukolsky S.A., Vetterling W.T., Flannery B.P. *Numerical Recipes in C. The Art of Scientific Computing*. Cambridge, Cambridge University Press, 1997, 1018 p.

17. Thompson Joe F., Warsi Z. A., Mastin C. V. *Numerical Grid Generation, Foundations and Applications*. Amsterdam, North-Holland, 1985, 331 p.

18. Dmitriev V.M., Gandzha T.V., Kochergin M.I. Matrix-topological analysis of component circuits, *Tomsk State University Journal of Control and Computer Science*, 2023, no. 62, pp. 25–35 (in Russ.).

19. Kolesov Yu.B., Senichenkov Yu.B. *Ob'ektno-orientirovannoe modelirovanie v srede Rand Model Designer 7* [Object-oriented modeling in the Rand Model Designer 7 environment]. Moscow, Prospect, 2016, 256 p. (in Russ.).

Maksim I. Kochergin

Candidate of Sciences in Engineering, Associate Professor, Department of Computer Systems in Control and Design, Tomsk University of Control Systems and Radioelectronics 40, Lenin pr., Tomsk, Russia, 634050
 ORCID: 0000-0001-7404-8718
 Phone: +7 (382-2) 70-15-15, int.: 20-64
 Email: maksim.i.kochergin@tusur.ru